

AD-A044 947 ROME AIR DEVELOPMENT CENTER GRIFFISS AFB N Y
A STATISTICS COLLECTION PACKAGE FOR THE JOVIAL J3 PROGRAMMING L--ETC(U)
F/G 9/2
SEP 77 R E STOVER
RADC-TR-77-293

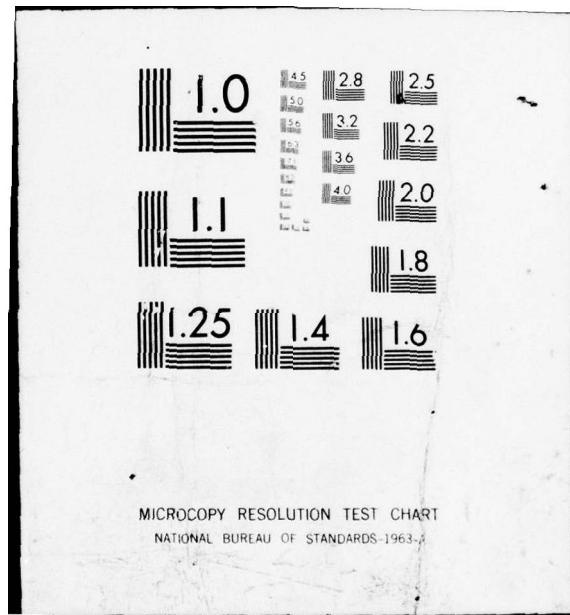
UNCLASSIFIED

NL

| OF /
AD
A044 947



END
DATE
FILED
11-77
DDC



AD A 044947

RADC-TR-77-293
In-house Report
September 1977

P.S.



A STATISTICS COLLECTION PACKAGE FOR THE JOVIAL J3 PROGRAMMING LANGUAGE

Robert E. Stover, Jr.

Approved for public release; distribution unlimited.

AD No. _____
DDC FILE COPY

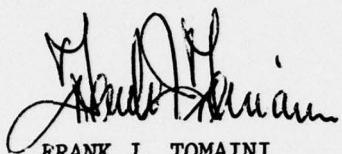
ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

D D C
REF ID: A
OCT 7 1977

This report has been reviewed by the Office of Information, RADC, and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

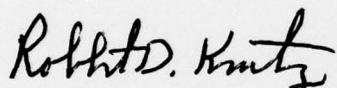
This report has been reviewed and is approved for publication.

APPROVED:



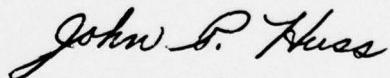
FRANK J. TOMAINI
Chief, Information Processing Branch
Information Sciences Division

APPROVED:



ROBERT D. KRUTZ, Col, USAF
Chief, Information Sciences Division

FOR THE COMMANDER:



JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (DAP) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

⑦ Rep't. for Jul 74 - Dec 76,

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADC-TR-77-293	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A STATISTICS COLLECTION PACKAGE FOR THE JOVIAL J3 PROGRAMMING LANGUAGE	5. TYPE OF REPORT & PERIOD COVERED In-house Jul 74 - Dec 76	6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) Robert E. Stover, Jr.	8. CONTRACT OR GRANT NUMBER(s) N/A	9. PERFORMING ORGANIZATION NAME AND ADDRESS Rome Air Development Center (ISIS) Griffiss AFB NY 13441
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62702F 55810002	11. REPORT DATE Sept 1977	12. CONTROLLING OFFICE NAME AND ADDRESS Same
13. NUMBER OF PAGES 40	14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same 16-5581 12-20	15. SECURITY CLASS. (of this report) Unclassified
16. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	17. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.	18. SUPPLEMENTARY NOTES None
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) High Order Language Usage Computer Programming Software Statistics	Hash Coding Symbol Table JOVIAL Programming Errors	20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report describes a software package developed in-house by RADC to measure counts, averages, and percentages relative to the usage of constructs and features of the JOVIAL J3 high order language by programmers. The numbers of occurrences of certain language features are obtained by processing the input JOVIAL J3 sources program in a manner similar to that employed by the front end of a compiler. This data is then used to calculate other quantities, averages, and percentages. Hash coding of identifiers in the input program and (Over)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

309050 K

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20.

any associated compools, and a symbol table in which information about these identifiers is recorded are used to assist in the statistics collection process.

This statistics collector is expected to be a valuable tool in the development of JOVIAL J3 and other programming languages by providing guidance relative to (1) more effective methods of programming, (2) implementation of compilers with greater efficiency, and (3) possible language changes.

This report also includes a summary of the syntax, semantics, and computer system interface errors made by the implementor in the process of development of the software package. The visibility provided by this information is expected to increase understanding of the nature, causes, and methods of avoidance of software errors.

ACCESSED BY

STN	WHITE SERVICE	<input checked="" type="checkbox"/>
INC	BILL SOUTHWAY	<input type="checkbox"/>
UNARMED		
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODES		
INCL	AVAIL AND OR SPECIAL	
A		

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

A STATISTICS COLLECTION PACKAGE FOR THE JOVIAL J3 PROGRAMMING LANGUAGE

Robert E. Stover, Jr.

1.0 INTRODUCTION

The JOVIAL J3 Statistics Collector, produced under an in-house effort by the Rome Air Development Center, is a software package designed to measure usage of the various constructs and features of the JOVIAL J3 higher-order computer programming language. The statistics collector takes as input a JOVIAL J3 source program, and operates upon this code to obtain various statistical quantities about the program. It is hoped that the statistical information derived by this package will enable more effective programming in the JOVIAL J3 language, greater efficiency in implementation of JOVIAL J3 compilers, and suggestions regarding possible language changes. Since JOVIAL J3 has been the official Air Force language for command and control applications, it is an ideal language for such analysis.

This statistics collector is currently hosted and running on the HIS 600/6000 GCOS computer system at RADC, but has been designed to be as system independent as possible. It processes JOVIAL J3 as described by MIL-STD-1588 (USAF), 30 June 1976, with certain features added for compatibility with the JOVIAL Compiler Implementation Tool (JOCIT).

A summary of basic facts about the JOVIAL J3 Statistics Collector appears in Table 1.

2.0 HISTORY

A brief summary of key events in the development of the JOVIAL J3 Statistics Collector is presented here. Since the productivity of the programmer and implementor, expressed in lines of code generated as a function of man-hours of work, is of interest, this information is presented in Graph 1.

2.1 Early Work

Construction of the JOVIAL J3 Statistics Collector began in July 1974, after a need for statistical information about the language became evident within the Air Force. The first task undertaken involved the

execution of a hash coding(*) routine with a given list of names as input; no processing of an actual program was performed.

In November 1974, a design for the statistics collector consisting of a compool, main program, routine to return tokens of the language, and output routine was put onto the system. At first the token returning routine was merely a prearranged list of tokens, but soon an actual operational scanner was implemented. These four modules correspond to the present STCOMP, STATCO, GETTOK, and PUTOUT.

2.2 Process of Major Development

Once its basic structure had been established, the statistics collector was progressively beefed up. At first the package only yielded information on numbers of types of statements and perhaps one or two other miscellaneous items, but in December 1974, a subroutine to process assignment statements and count various operators and operands contained therein was implemented. In January 1975, hash coding of identifiers was restored as part of the package, using the independently compiled subroutine TOKSH.

Work on the statistics collector was then suspended for six months. One of the first tasks conducted upon resumption of the effort was the detection and counting of various types of declarations. Also, the recompilation of the entire statistics collector every time a change was made to any part of it was found to be wasteful and expensive, so a procedure was established for saving the object code of modules on permfiles on the GCOS computer system and only recompiling modules affected by changes prior to a run. Since changes to a compool generally require recompilation of any referencing programs, it was decided to move the statistical data, only referenced by STATCO and PUTOUT, into a new compool COMPST, requiring recompilation of only this compool and these two executable modules whenever the statistical data base was revised.

In October 1975, a symbol table containing information about identifiers was incorporated into the statistics collector, and this capability has

(*) Hash coding is a method by which a name is assigned to a unique location in a table having n entries by determining the value modulo n of some integer function of the name, such as the sum of the machine representations of the words containing it. If the entry determined for a given name is already occupied by another name (this is called a collision), a quadratic function is applied with successive integers, and this amount added to the original location, to determine a new table entry until a vacant position for the name can be found in the table. When this name is again referenced by the program, this procedure can be repeated to find where it was originally stored, precluding the necessity of searching through most of the table for the name.

been progressively upgraded to the present. The hash coding output routine, previously called HSHOUT, was renamed TABOUT, and now printed both the hash coding and symbol tables.

The inclusion of a symbol table prompted the revision of the statistics collector to a two-pass structure. This was done for a while by making GETTOK the entire first pass, fetching all of the tokens upon one call from STATCO and creating the symbol table, while STATCO performed the second pass statistical processing. This was found to be inefficient, so a new subroutine module, PASS1, was called once from STATCO, and called GETTOK, which had been reverted to yielding one token per call, whenever the next token in the input stream was needed. At this point, all eight independently compiled modules of the present statistics collector structure were in place (see Section 3.0).

From this point until major testing of the package was begun in June 1976, most of the work consisted of refinement of the statistics collector and increasing the types of statistical quantities obtained. Two major extensions which deserve mention are the implementation of compool resolution of identifiers and the processing of define directives and substitutions.

2.3 Recent Testing and Debugging

During most of the process of its development, the statistics collector was run with its own modules as input. However, during June, July, and December 1976, the JOVIAL J3 Compiler Validation System (JCVS-J3) and certain software produced by Strategic Air Command (SAC) headquarters were used as input to the package. Previously undetected errors uncovered by this process included failure to handle unnamed tables; wrong detection of control transfer and switch names in some instances; and incorrect symbol table type assignments from item descriptions. The utility of the SAC software was limited by lack of access to the compools referenced by those programs.

3.0 DESCRIPTION OF MODULES

The JOVIAL J3 Statistics Collector currently consists of eight independently compiled modules: two compools, one driver program, and five procedures performing various functions within the package. Reference has already been made in Sections 2.1 and 2.2 to the historical development of each of these modules. They are individually described below, and their logical interrelationship appears in Table 2. Table 3 lists the size of each module, and summarizes its function.

3.1 STCOMP

STCOMP is the main compool of the statistics collector. It contains declarations of simple items, tables, arrays, files, and procedures used

during execution, as well as a few declarations of statistical items calculated by the scanner GETTOK. It is referenced by all executable modules.

3.2 COMPST

COMPST is the compool containing declarations of most statistical quantities. It is only referenced by STATCO, where these values are calculated, and PUTOUT, where they are output.

3.3 STATCO

STATCO is the driver program of the package. It calls PASS1 to perform the first pass, then does the statistical counting and calculation of the second pass itself. After this is completed, it calls output routines PUTOUT and TABOUT. It contains closes to collect statistics about arithmetic expressions, named tables, unnamed tables, and arrays, and subroutines to determine type of assignment and to collect statistics about item description types.

3.4 PASS1

PASS1 conducts the first pass of statistics collection, which consists primarily of symbol table creation. It calls GETTOK to obtain the next input token when needed, employing token lookahead in some instances. It also calls TOKHSH to hash code identifiers. PASS1 contains a close for assigning symbol table name, scope, and class to identifiers, and subroutines to discern between unsigned and signed integer and fixed items and to create symbol table entries for like tables.

3.5 GETTOK

GETTOK is the scanner. It performs essentially as a finite state machine, scanning the stream of input characters until a token of the language is detected. In the case of a define name reference, the scan can be diverted to the appropriate define string. GETTOK contains subroutines for assigning a numerical value to each character and for detecting and handling the end of an input line or define string.

A list of the tokens of the JOVIAL J3 language, all of which are now detected by GETTOK, appears in Table 4.

3.6 TOKHSH

TOKHSH, the hash coding routine, uses a quadratic hash coding algorithm which places each distinct compool and source program identifier at a unique position in the hash table, and sets up a symbol table reference for the identifier. It contains subroutines to determine whether identifiers of identical spelling are in fact the same or different with regard to scope and context, and to determine how an identifier is declared.

3.7 PUTOUT

PUTOUT is called by STATCO, and prints a summary of the statistical information collected and evaluated by the package.

3.8 TABOUT

TABOUT, also called by STATCO, prints the hash coding and symbol tables for the input program, including any associated compools.

4.0 DOCUMENTATION

Written information about the JOVIAL J3 Statistics Collector and its construction and implementation has been maintained in various forms for a number of aspects of the effort. These are listed below, and their structure is portrayed in Table 5. The number of pages in each document appears in Table 6.

4.1 Descriptions and Reports

A considerable amount of general information about the statistics collector has been provided at various times during the course of the project. A brief summary of such documents is presented here.

Early in the effort, a brief review of the nature and purpose of the statistics collector was set forth in "JOVIAL J3 Statistics Collector," dated 10 September 1974. This was accompanied by "Statistics Collector Components," an outline of statistics expected to be gathered by the package, bearing the same date. This report and outline both reside on the HIS 600/6000 Multics computer system at RADC.

An early, and now somewhat out of date, design plan for the statistics collector appears in "Overall Design of the JOVIAL J3 Statistics Collector." The date of this paper is uncertain, but is probably around October 1974. A handwritten copy of this report exists in the statistics collector records, but it has never been put onto a computer system.

Two interim reports of progress in the development of the statistics collector have been produced. The first, "Status of the JOVIAL J3 Statistics Collector," dated 13 January 1975, resides on the GCOS computer system at RADC, and has been distributed to some extent. This report summarizes early work on the package, discusses its structure in some detail, and provides an analysis of errors made up to that point. This material was updated in "Second Interim Report on the JOVIAL J3 Statistics Collector," dated 26 March 1976, which resides on the Multics computer system at RADC, but has never been formally published.

An RADC Status Report, Form 77b, dated 1 October 1976, describes the state of the statistics collector effort as of that date in handwritten

tabular form, and is in the statistics collector records.

It is expected that a brief report will be produced at the end of the entire statistics collector project, updating this RADC Technical Report and summarizing the final work on the effort.

4.2 List of Highlights

This is a chronological listing of the major milestones and extensions of capability for the statistics collector from its inception to the present.

4.3 Daily Diary

This is a day-by-day account of work performed on the statistics collector, and other related events, from 23 September 1974 to the present. Monthly summaries describe work accomplished prior to the inception of daily accounting.

4.4 Time and Cost Information

Hours spent working on the statistics collector, number of terminal sessions, total terminal connect time, terminal cost, number of batch runs, total processor time for batch runs, batch cost, and total computer cost are recorded for each date on which work was performed on the statistics collector from 15 October 1974 to the present. Table 7 contains a summary of this information on a monthly basis.

4.5 Error Data

This is a record of all programmer errors made on the package since 1 October 1974, and all compiler and system errors occurring since its inception. Prior to 30 September 1975, accounts of errors were written up in paragraph form; since 1 October 1975, they have been recorded in tabular form. Programmer errors have been subdivided into six categories: forgetfulness, logic, data management, subroutine linkage, input/output, and resource allocation. Aspects of errors considered include criticality, relationship to attempted correction of previous errors, and number of runs required to fix. A summary of the error information obtained so far in the project appears in Table 8.

Reclassification of these errors based on the categories established by TRW Systems Group in a recent study of software reliability (1) has been accomplished. The results of this reclassification appear in Table 9. Further discussion of error analysis appears in Section 6.

4.6 Computer Output

Listings of terminal sessions and batch runs have been saved since quite early in the effort. These listings are not complete, but cover most of the significant work done on the statistics collector.

5.0 RESULTS ACHIEVED TO DATE

As is indicated in Table 10, the statistics collector now possesses an extensive capability for describing usage of JOVIAL J3 constructs and features by programmers. Counts and percentages of both declaration and executable statement types are provided. A further breakdown of tokens in arithmetic expressions is derived. Information such as number of lines in the program, average line length, average comment length, etc., is also calculated.

No real analysis of the statistical data obtained by running various programs through the statistics collector has yet been conducted. In fact, some of the programs used as input are not truly representative of JOVIAL J3 software, so the applicability of such an analysis might be limited. However, it has been noticed that the use of structured programming technology has been somewhat lacking (data not explicitly declared, IF statements used in preference to IFEITH, many GOTO statements, etc.) Another observation is that certain programmers tend to completely avoid using particular features of the language, such as arrays or exchange statements.

An example of an actual output listing from execution of the statistics collector, including hash coding and symbol tables, appears in Table 11. The first pass and symbol table routine PASS1 of the statistics collector itself, with reference to main statistics collector compool STCOMP, was used as source program input in this case.

6.0 ANALYSIS OF ERRORS

As mentioned in Section 4.5, the source and compiler errors made in the process of coding the statistics collector have been categorized both in terms of the implementor's own classification scheme, and that devised by TRW Systems Group (2). The reason for going to such great lengths to analyze these errors is that it was felt that this effort provided a ready-made opportunity to observe the commission, discovery, and correction of errors during the actual software development process. It is hoped that investigation of this data and its comparison with that from other software projects will provide meaningful insight into the problem of software errors.

As indicated previously, Tables 8 and 9 summarize this data. A further analysis of each of the two error classifications follows in Sections 6.1 and 6.2.

Another important aspect of source errors is the relationship between the total number of errors and the lines of code produced. Graph 2 expresses this relationship.

6.1 Implementor's Classification

As stated in Section 4.5, this classification consists of six categories of user errors, plus a separate category for compiler or system errors. The most prevalent categories of user errors are seen to be logic and data management. This reflects the extensive logical branching required to identify language constructs from the source code and tokens, and the extensive manipulation required to operate upon data entities, particularly in relation to the symbol table. Subroutine linkage errors are fewer in number, but are more often critical (causing all or most of the statistics collector to fail to operate properly), as would be expected. Resource allocation errors are almost always critical, for the package cannot really run without the provision of required space or time. Nearly 20% of all user errors are related to attempts to correct previous errors; this is considered a high figure.

6.2 TRW Classification

As is seen in Table 9, the errors are divided among 20 categories (3),⁴ of which are considered non-applicable to this effort for the reasons stated. As with the previous classification, logic and data handling are the most prevalent error categories, and errors related to the global operation of the software tend to be more critical. The "Recurrent" category is a catch-all for all errors resulting from another attempted error correction, and these errors are not further classified, except with regard to criticality. Compiler or system errors are generally included in the "Operating System/System Support Software" category.

The total number of errors included in this second classification is slightly greater than that in the first because some errors were made between the times of the two classifications, and also because some single errors in the first classification were considered multiple errors for purposes of the second.

7.0 FUTURE PLANS

The statistics collector is now pretty well completed, with most of the desired capability having been incorporated into the package. There are a few features yet to be added. These include collection of further information about IF, IFEITH, and FOR statements, and the levels of nesting of such statements; increased analysis of the right side of assignment statements; treatment of arithmetic expressions in contexts besides assignment; and recording of frequency of usage of implementation procedures and functions.

It is also planned to modify the statistics collector to record information about a large number of programs in a data base maintained by the host system. Currently, information can only be obtained about one program at a time, and is lost after output. By replacing the process of outputting information about a single program with that of using this information to update a continuously existing data base, a

statistical summary of as many programs as desired can be maintained.

8.0 CONCLUSION

As has already been indicated, the JOVIAL J3 Statistics Collector has been in the process of development for some time, and is now approaching completion. The value of this tool to the Air Force and others cannot be fully appraised until it is run against a significant number of representative yet diverse JOVIAL J3 programs. However, the specific results derived from the limited inputs applied to the package to date give promise of its increased utility in the future. The task of its construction has been a valuable learning experience for the implementor, both in regard to the JOVIAL J3 language and to software design and development procedures. As pointed out in Sections 4.5 and 6.0, the software error information collected throughout most of this effort is expected to significantly increase understanding in this area.

In short, the JOVIAL J3 Statistics Collector has already enhanced a number of aspects of software understanding within the Air Force, and will do so in even greater measure in the future. It will prove to be a highly useful tool in assisting the Air Force in various phases of JOVIAL J3 and software development.

REFERENCES

- (1) TRW Systems Group, Software Reliability Study, Technical Report, RADC-TR-76-238, Redondo Beach, CA, August 1976, p. 3-1 to 3-22.
- (2) Ibid.
- (3) Ibid., p. 3-3 to 3-8.

Start date: July 1974

Periods during which work conducted:

July 1974 through January 1975

July 1975 through July 1976

December 1976 through January 1977

Total time expenditure: 7 man/months

Total cost of computer (HIS 600/6000 GOOS) usage since 15 Oct 74:
\$7366.82

Number of independently compiled modules: 8

Lines of source code as of 21 Dec 76: 4023

User coding and implementation errors, 1 Oct 74 - 21 Dec 76: 285

Pages of written documentation as of 21 Dec 76: 307

Approximate percentage of code employing top-down design: 90%

Approximate percentage of code employing structured programming: 95%

Percentage of code written in JOVIAL J3: 100%

Table 1: Facts about JOVIAL J3 Statistics Collector

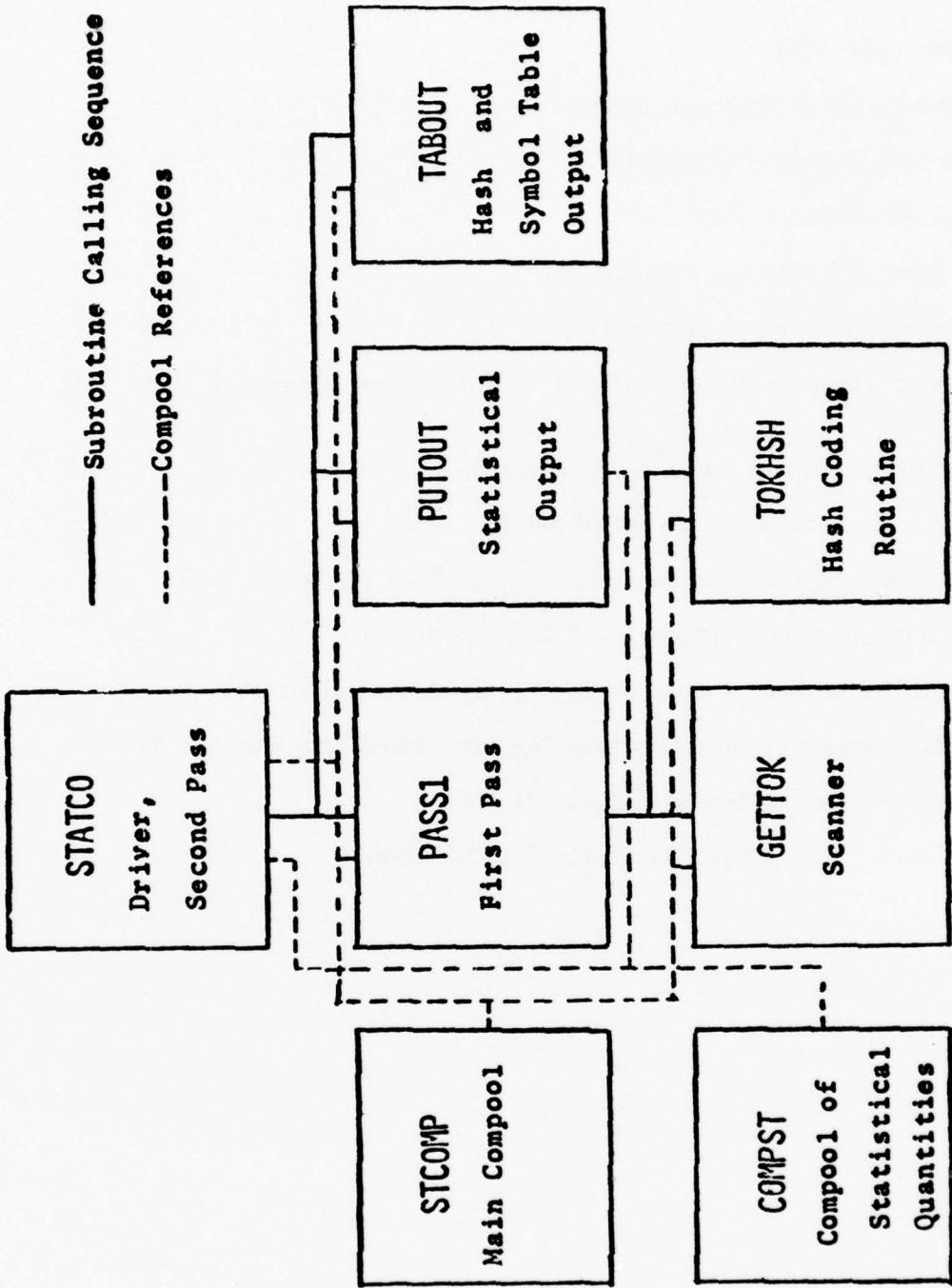


Table 2. Interconnection of Statistics Collector Modules

Module	Function	Size in Lines
STCOMP	Main compool	226
COMPST	Compool containing statistical data	213
STATOO	Driver program	810
PASS1	First pass, symbol table creation	931
GETTOK	Scanner, token detector	754
TOKHSH	Hash coding	287
PUTOUT	Statistical output	577
TABOUT	Hash coding and symbol table output	<u>225</u>
Total number of lines		4023
Average number of lines per module		503

Table 3: Size and Function of Statistics Collector Modules

I. Tokens identifying type of statement or clause

IF	FOR	RETURN
IFEITH	GOTO	TEST
ORIF	STOP	ASSIGN

II. Tokens identifying type of declaration

ITEM	CLOSE
STRING	PROGRAM or 'PROGRAM (independently compiled close)
TABLE	ARRAY
FILE	COMMON
PROC	MODE
OVERLAY	SWITCH
DEFINE	MONITOR

III. Single letter tokens indicating declaration attributes

A (fixed item, or integer if precision not specified)

B (boolean item or binary file)

C (ASCII item)

D (dense table packing)

F (floating item)

H (hollerith item or file)

I (integer item)

L (like table)

M (medium table packing)

N (no table packing)

P (item preset or parallel table structure)

R (rounding, rigid table size, or fixed length file)

S (signed or status item, or serial table structure)

T (transmission code item)

U (unsigned item)

V (variable table size, or variable length file)

IV. Statement or character string delimiters

\$ (statement or declaration terminator)

BEGIN, END (compound statement, grouped declaration, or constant list delimiters)

START, TERM (independent compilation delimiters)

DIRECT, JOVIAL (direct code delimiters)

() (parentheses)

(\$ \$) (subscript or BIT or BYTE parameter delimiters)

, (comma)

... (range limit separator in item descriptions)

. (period, statement name terminator)

V. Arithmetic, relational, or logical operators

A. Arithmetic and assignment operators

=	*
== (exchange)	/
+	** (exponentiation)
-	(* *) (exponentiation brackets)

B. Relational operators

EQ (equal)	LQ (less than or equal)
NQ (not equal)	GR (greater than)
LS (less than)	GQ (greater than or equal)

C. Logical operators

AND
OR
NOT

VI. Variables and constants

Identifier
FOR parameter
Integer constant

Hollerith constant
Transmission code constant
ASCII constant

Fixed constant	Status constant
Floating constant	Accumulator specification
Octal constant	

VII. Built-in functions

BIT	ENT or ENTRY
BYTE	LOC or 'LOC
CHAR	NENT
MANT	NWDSEN
ABS	ODD
(/ /) (absolute value brackets)	POS
ALL	

VIII. Input/output primitives

OPEN	INPUT
SHUT	OUTPUT

Table 4: Tokens of the JOVIAL J3 Language

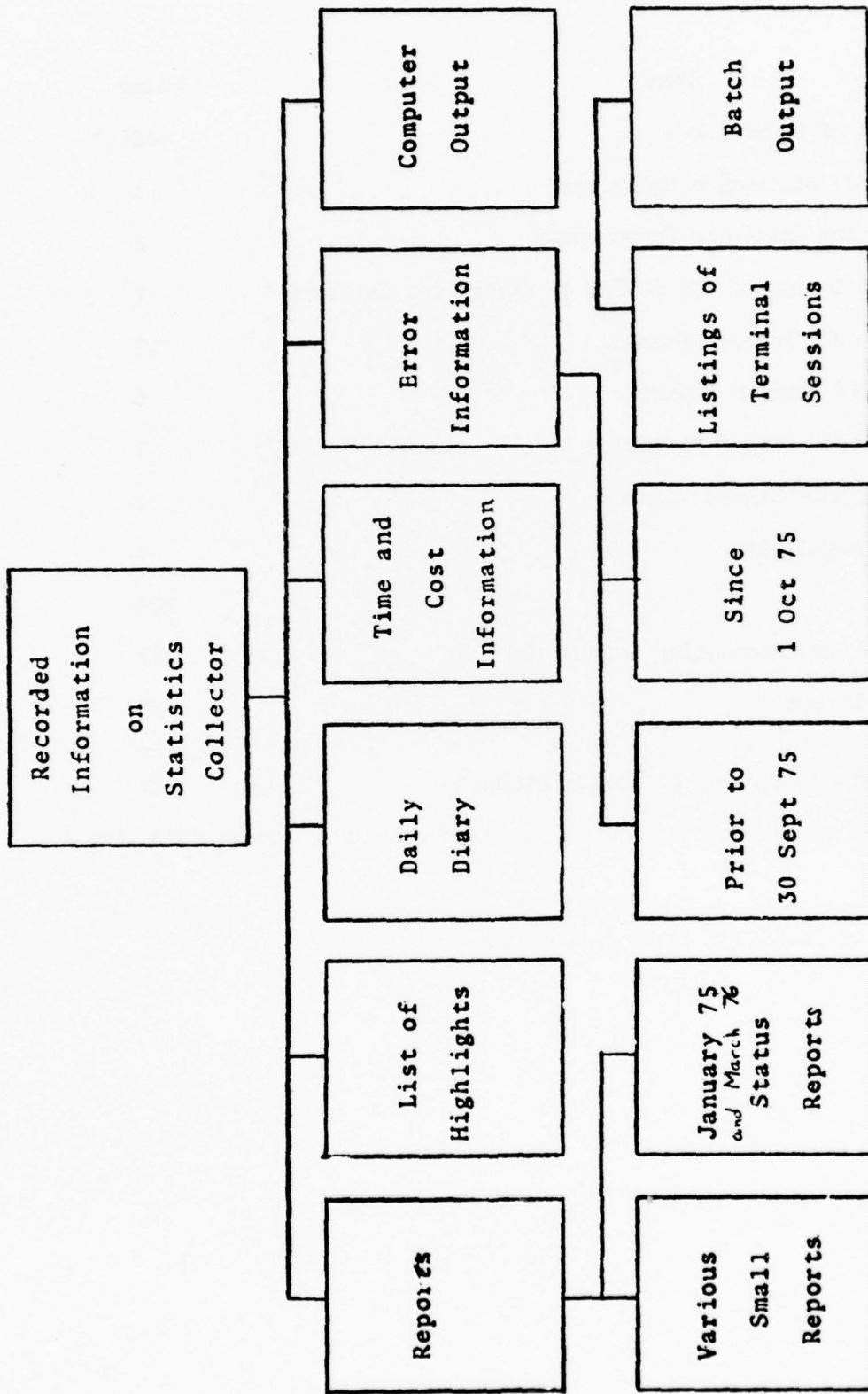


Table 2. Structure of Documentation about Statistics Collector

Item	Pages
Listings of Source Code	80(1)
"JOVIAL J3 Statistics Collector"	1
"Statistics Collector Components"	2
"Overall Design of the JOVIAL J3 Statistics Collector"	2
January 1975 Interim Report	13
March 1976 Interim Report	6
Early Sample Output Listing	1
Standard RADC Status Report	2
List of Highlights	4
Diary	108
Time and Cost Accounting Information	12
List of Errors	80
 Total Number of Pages of Documentation	 311

(1) Based on 50 lines of source code per page (code totals 4023 lines).

Table 6: Available Written Documentation Regarding the JOVIAL J3 Statistics Collector as of 21 December 1976

Month	Hours Spent	Terminal Usage			#	Batch Runs	Total Cost	
		Sessions	Time(1)	Cost		Time(2)	Cost	
October(3)	46	8	.812	\$6.52	8	.1113	\$60.68	\$67.20
November	86	30	9.464	63.31	26	.1585	167.70	231.01
December	44	21	6.465	45.27	17	.2792	212.27	257.54
1974 Totals(4)	176	59	16.741	\$115.10	51	.5490	\$440.65	\$555.75
January	51	13	4.633	\$50.21	8	.3740	\$293.49	\$343.70
February	0	0	.000	.00	0	.0000	.00	.00
March	0	0	.000	.00	0	.0000	.00	.00
April	0	0	.000	.00	0	.0000	.00	.00
May	0	0	.000	.00	0	.0000	.00	.00
June	0	0	.000	.00	0	.0000	.00	.00
July	28	9	3.689	26.45	14	.1554	96.94	123.39
August	63	37	9.172	69.70	32	.5506	370.75	440.45
September	81	63	12.880	99.88	49	.6800	445.21	545.09
October	76	22	8.728	65.38	19	.5982	224.05	289.43
November	48	15	3.579	30.45	11	.4132	144.06	174.51
December	102	46	17.298	136.91	41	3.2360	933.64	1070.55
1975 Totals	449	205	59.979	\$478.98	174	6.0074	\$2508.14	\$2987.12
January	80	23	11.400	\$85.57	20	.9993	\$290.54	\$376.11
February	63	30	10.497	82.28	29	.4693	556.65	638.93
March	73	22	5.876	49.05	21	.3107	432.19	481.24
April	44	25	9.657	73.92	21	.3221	280.10	354.02
May	61	20	9.402	68.34	17	.2279	232.09	300.43
June	125	109	29.610	239.14	77	1.2700	940.82	1179.96
July	9	4	1.725	12.36	8	.0576	46.75	59.11
August	0	0	.000	.00	0	.0000	.00	.00
September	0	0	.000	.00	0	.0000	.00	.00
October	0	0	.000	.00	0	.0000	.00	.00
November	0	0	.000	.00	0	.0000	.00	.00
December	73	38	10.968	101.65	33	.4146	332.50	434.15
1976 Totals	528	271	89.135	\$712.31	226	4.0715	\$3111.64	\$3823.95
Grand Totals	1153	535	165.855	\$1306.39	451	10.6279	\$6060.43	\$7366.82

(1) Clock time.

(2) Central Processing Unit (CPU) time.

(3) From 15 October 1974 through end of month.

(4) From 15 October 1974 through end of year.

Table 7: Statistics Collector Time and Cost Expenditures,
15 October 1974 - 31 December 1976

User Errors

Category	Total # of Errors	Critical Errors(1) #	%	Secondary Errors(2) #	%	% of all User Errors
Forgetfulness	49	28	57.14	6	12.24	17.19
Logic	119	22	18.49	22	18.49	41.75
Data Management	80	35	43.75	16	20.00	28.07
Subroutine Linkage	19	13	68.42	7	36.84	6.67
Input/Output	10	1	10.00	1	10.00	3.51
Resource Allocation	8	8	10.00	0	0.00	2.81
Totals	285	107	37.54	52	18.25	100.00

Compiler or System Errors(3)

Critical	13
Non-critical	<u>4</u>
Total	17

(1)Errors causing most or all of the statistics collector to fail to operate properly.

(2)Errors caused by attempted correction of another error.

(3)Collection of data begun at inception of project in July, 1974.

Table 8: Statistics Collector Errors from 1 October 1974 through 21 December 1976, Using Implementor's Own Classification

Error Cate- gory	Description of Category	Number of Errors			% of Critical Errors	% of Total # of Errors
		Non- Critical	Critical	Total		
1	Computational	5	22	27	18.52	7.92
2	Logic	29	78	107	27.10	31.38
3	Input/Output	5	10	15	33.33	4.40
4	Data Handling	26	34	60	43.33	17.60
5	Operating System/ System Support Software	13	4	17	76.47	4.99
6	Configuration	0	0	0	0.00	0.00
7	Routine/ Routine Interface	8	7	15	53.33	4.40
8	Routine/ System Software Interface	2	3	5	40.00	1.47
9	Tape Processing Interface	Not applicable because magnetic tapes not directly used with software				
10	User Interface	2	0	2	100.00	0.59
11	Data Base Interface	0	0	0	0.00	0.00
12	User Requested Changes	Not applicable because not yet any other users				
13	Preset Data Base	10	2	12	83.33	3.52
14	Global Variable/ Compool Definition	7	2	9	77.78	2.64
15	Recurrent	26	38	64	40.62	18.77
16	Documentation	Not applicable because such errors not recorded				
17	Requirements Compliance	8	0	8	100.00	2.35
18	Operator	0	0	0	0.00	0.00
19	Questions	Not applicable because not yet any other users				
20	Unidentified	0	0	0	0.00	0.00
Totals		141	200	341	41.35	100.00

Table 9: Statistics Collector Errors from 1 October 1974 through 31 December 1976, Using TRW Systems Group Classification

I. General information about program

- A. Number of lines
- B. Number of tokens with and without define expansion
- C. Number of characters, and average per line
- D. Number of comments, and average length
- E. Number of define directives, and average length
- F. Number of define calls, and average per define directive

II. Number and percentage of various types of declarations

- A. Simple item
- B. Array, including breakdown by number and size of dimensions
- C. Table, including breakdown by table attributes
- D. Overlay
- E. File
- F. Switch
- G. Close
- H. Program
- I. Common
- J. Monitor
- K. Procedure and function
- L. Breakdown of all item descriptions by type
- M. Number of mode directives

III. Summary of external references

- A. Breakdown of compool resolved names
- B. External closes referenced
- C. System defined procedures and functions
- D. Mode defined simple items
- E. Simple items resolved to default attributes
- F. Percentage of identifiers declared implicitly

IV. Number and percentage of various types of statements

- A. Assignment, including breakdown by type of assignment
- B. Exchange
- C. GOTO
- D. Return
- E. Stop
- F. Test
- G. Procedure call
- H. Input/output, including breakdown by exact nature
- I. IF

J. IFEITH
K. FOR
L. Direct

V. Summary information about statements

- A. Total number of simple and complex executable statements
- B. Total number of executable and non-executable statements
- C. Number of compound statements
- D. Number of statement labels

VI. Information about arithmetic expressions

- A. Total number
- B. Total number of tokens
- C. Number of constants
- D. Number of variables
- E. Number of subscripted variables
- F. Counts of occurrences of various operators
- G. Counts of references to built-in and other functions

Table 10: Statistical Information about Source Program Currently Available

STATISTICAL SUMMARY OF JOVIAL J3 LANGUAGE USAGE

NUMBER OF LINES IN INPUT SOURCE PROGRAM = 931
 NUMBER OF TOKENS IN INPUT PROGRAM EXCLUDING DEFINE STRING EXPANSION = 4143
 NUMBER OF TOKENS IN INPUT PROGRAM INCLUDING FULL DEFINE STRING EXPANSION = 4143
 NUMBER OF CHARACTERS IN INPUT SOURCE PROGRAM INCLUDING LEADING BLANKS AND COMMENTS = 42095
 AVERAGE NUMBER OF CHARACTERS PER LINE IN INPUT PROGRAM INCLUDING LEADING BLANKS AND COMMENTS = 45.21
 NUMBER OF COMMENTS IN INPUT PROGRAM = 556
 AVERAGE LENGTH IN CHARACTERS OF COMMENTS IN INPUT PROGRAM = 32.51
 NUMBER OF DEFINE DIRECTIVES IN INPUT SOURCE PROGRAM = 0
 AVERAGE LENGTH IN CHARACTERS OF INPUT DEFINE DIRECTIVES = 0.
 NUMBER OF DEFINE CALLS IN INPUT SOURCE PROGRAM = 0
 AVERAGE NUMBER OF DEFINE CALLS PER DEFINE DIRECTIVE = 0.

 NUMBER OF DECLARATIONS = 42
 NUMBER OF SIMPLE ITEM DECLARATIONS = 38
 PERCENTAGE OF ALL DECLARATIONS = 90.48 %
 NUMBER OF ARRAY DECLARATIONS = 0
 PERCENTAGE OF ALL DECLARATIONS = 0. %
 AVERAGE NUMBER OF DIMENSIONS PER ARRAY = 0.
 DISTRIBUTION OF NUMBERS OF ARRAY DIMENSIONS:
 1 DIMENSION - 0 5 DIMENSIONS - 0
 2 DIMENSIONS - 0 6 DIMENSIONS - 0
 3 DIMENSIONS - 0 7 DIMENSIONS - 0
 4 DIMENSIONS - 0
 AVERAGE OF ALL ARRAY DIMENSION SIZES = 0.

 NUMBER OF TABLE DECLARATIONS = 0
 PERCENTAGE OF ALL DECLARATIONS = 0. %
 TOTAL NUMBER OF TABLE ITEM AND STRING ITEM DECLARATIONS IN ALL TABLES = 0
 0 TABLE ITEMS, 0 STRING ITEMS
 AVERAGE NUMBER OF ITEMS PER TABLE = 0.
 AVERAGE NUMBER OF ENTRIES PER TABLE = 0.
 AVERAGE NUMBER OF WORDS PER SPECIFIED TABLE ENTRY = 0.
 DISTRIBUTION OF TABLE SIZES BY NUMBER OF ENTRIES:
 10 OR FEWER ENTRIES 0 1,001 TO 10,000 ENTRIES 0
 11 TO 100 ENTRIES 0 MORE THAN 10,000 ENTRIES 0
 101 TO 1,000 ENTRIES 0
 TYPE OF ENTRY: O ORDINARY, O SPECIFIED
 NATURE OF NUMBER OF ENTRIES: O RIGID, O VARIABLE
 STRUCTURE: O PARALLEL, O SERIAL
 PACKING: O NO, O MEDIUM, O DENSE
 NUMBER OF LIKE TABLES = 0
 NUMBER OF UNNAMED TABLES = 0
 TOTAL NUMBER OF DECLARATIONS HAVING EXPLICIT OR IMPLICIT ITEM DESCRIPTION = 39
 INTEGER 14
 UNSIGNED 0
 SIGNED 14
 FIXED POINT 0
 UNSIGNED 0
 SIGNED 0
 NUMBER OF OVERLAY DECLARATIONS = 0
 PERCENTAGE OF ALL DECLARATIONS = 0. %
 NUMBER OF FILE DECLARATIONS = 0
 PERCENTAGE OF ALL DECLARATIONS = 0. %

BEST AVAILABLE COPY

NUMBER OF SWITCH DECLARATIONS = 0
PERCENTAGE OF ALL DECLARATIONS = 0. %
NUMBER OF PROGRAM DECLARATIONS = 0
PERCENTAGE OF ALL DECLARATIONS = 0. %
NUMBER OF CLOSE DECLARATIONS = 1
PERCENTAGE OF ALL DECLARATIONS = 2.38 %
NUMBER OF PROCEDURE AND FUNCTION DECLARATIONS = 3
PERCENTAGE OF ALL DECLARATIONS = 7.14 %
NUMBER OF PROCEDURE DECLARATIONS = 2
PERCENTAGE OF ALL DECLARATIONS = 7.14 %
FUNCTION DECLARATIONS = 1
NUMBER OF COMMON DECLARATIONS = 0
PERCENTAGE OF ALL DECLARATIONS = 0. %
NUMBER OF MONITOR DECLARATIONS = 0
PERCENTAGE OF ALL DECLARATIONS = 0. %
NUMBER OF MODE DIRECTIVES = 0
NUMBER OF DEFINE DIRECTIVES = 0

NUMBER OF NAMES RESOLVED BY COMPOOL REFERENCE = 71
SIMPLE ITEMS = 39
ARRAYS = 5
TABLES = 0
NUMBER OF EXTERNAL CLOSSES REFERENCED = 0
NUMBER OF SYSTEM DEFINED PROCEDURES AND FUNCTIONS = 1
NUMBER OF MODE DEFINED SIMPLE ITEMS = 0
NUMBER OF SIMPLE ITEMS RESOLVED TO DEFAULT ATTRIBUTES = 0
PERCENTAGE OF DATA, PROCEDURE, SWITCH, AND CLOSE IDENTIFIERS LACKING EXPLICIT PROGRAM OR COMPOOL DECLARATION = 0.88

25 NUMBER OF ASSIGNMENT STATEMENTS = 241
PERCENTAGE OF ALL SIMPLE STATEMENTS = 87.64 %
NUMBER OF ARITHMETIC ASSIGNMENTS = 88
NUMBER OF CHARACTER ASSIGNMENTS = 20
NUMBER OF BOOLEAN ASSIGNMENTS = 41
NUMBER OF STATUS ASSIGNMENTS = 92
NUMBER OF ENTRY ASSIGNMENTS = 0
PERCENTAGE OF ALL SIMPLE STATEMENTS = 0. %
NUMBER OF GOTO STATEMENTS = 8
PERCENTAGE OF ALL SIMPLE STATEMENTS = 2.91 %
NUMBER OF RETURN STATEMENTS = 3
PERCENTAGE OF ALL SIMPLE STATEMENTS = 1.09 %
NUMBER OF STOP STATEMENTS = 0
PERCENTAGE OF ALL SIMPLE STATEMENTS = 0. %
NUMBER OF TEST STATEMENTS = 0
PERCENTAGE OF ALL SIMPLE STATEMENTS = 0. %
NUMBER OF PROCEDURE CALLS = 21
PERCENTAGE OF ALL SIMPLE STATEMENTS = 7.64 %
NUMBER OF INPUT/OUTPUT STATEMENTS = 2
PERCENTAGE OF ALL SIMPLE STATEMENTS = 0.73 %
NUMBER OF FILE OPENING STATEMENTS WITHOUT DATA TRANSMISSION = 0
NUMBER OF FILE OPENING STATEMENTS WITH DATA TRANSMISSION = 0
NUMBER OF FILE CLOSING STATEMENTS WITHOUT DATA TRANSMISSION = 0
NUMBER OF FILE CLOSING STATEMENTS WITH DATA TRANSMISSION = 0
NUMBER OF INPUT STATEMENTS = 2
NUMBER OF OUTPUT STATEMENTS = 0

TOTAL NUMBER OF SIMPLE STATEMENTS IN PROGRAM = 275

NUMBER OF IF STATEMENTS = 46
PERCENTAGE OF ALL COMPLEX STATEMENTS = 52.27 %

NUMBER OF IF/EITH STATEMENTS = 32
PERCENTAGE OF ALL COMPLEX STATEMENTS = 36.36 %

NUMBER OF FOR STATEMENTS = 10
PERCENTAGE OF ALL COMPLEX STATEMENTS = 11.36 %

NUMBER OF DIRECT STATEMENTS = 0
PERCENTAGE OF ALL COMPLEX STATEMENTS = 0. %

TOTAL NUMBER OF COMPLEX STATEMENTS IN PROGRAM = 88

TOTAL NUMBER OF SIMPLE AND COMPLEX EXECUTABLE STATEMENTS IN PROGRAM = 363
TOTAL NUMBER OF EXECUTABLE AND NON-EXECUTABLE STATEMENTS IN PROGRAM = 405

NUMBER OF COMPOUND STATEMENTS = 87

NUMBER OF STATEMENT LABELS = 7

TOTAL NUMBER OF ARITHMETIC EXPRESSIONS = 88

TOTAL NUMBER OF TOKENS IN ARITHMETIC EXPRESSIONS = 199

NUMBER OF CONSTANTS = 66

NUMBER OF VARIABLES = 75

NUMBER OF SUBSCRIPTED VARIABLES = 3

NUMBER OF PLUS SIGNS = 35
NUMBER OF MINUS SIGNS = 9

NUMBER OF MULTIPLICATION SIGNS = 4
NUMBER OF DIVISION SIGNS = 4

NUMBER OF EXPONENTIATION OPERATORS = 0
NUMBER OF ABSOLUTE VALUE REFERENCES = 0

NUMBER OF USES OF BIT MODIFIER = 0
NUMBER OF USES OF BYTE MODIFIER = 0

NUMBER OF USES OF CHAR MODIFIER = 0
NUMBER OF LOC FUNCTION CAL'S = 0

NUMBER OF USES OF MANT MODIFIER = 0
NUMBER OF USES OF NEXT MODIFIER = 0

NUMBER OF NWDSEN FUNCTION CALLS = 0
NUMBER OF USES OF PDS MODIFIER = 0

NUMBER OF OTHER FUNCTION CALLS = 0

HASH CODING TABLE

0	1	2	3	4	5	6	7	8	9
5	10	11	12	13	14	15	16	17	18
15	15	17	17	18	19	21	21	23	24
20	20	22	22	23	24	25	26	28	29
25	25	27	27	28	29	29	30	32	33
30	30	31	31	32	33	33	34	34	34
35	35	36	36	37	38	38	39	39	39
40	40	41	42	43	43	44	44	44	44
45	45	46	47	48	48	49	49	49	49
50	50	CURCON	51	52	53	54	54	55	55
55	55	56	56	57	58	59	59	59	59
60	60	61	61	62	63	64	64	64	64
65	65	66	66	67	68	69	69	69	69
70	70	71	71	72	73	74	74	74	74
75	75	76	76	77	78	79	79	79	79
80	80	81	81	82	83	84	84	84	84
85	85	ACMLGH	86	87	88	89	89	89	89
90	90	NLINES	91	92	93	94	94	94	94
95	95	96	96	97	98	99	99	99	99
100	100	101	101	102	103	104	104	104	104
105	105	106	106	107	108	109	109	109	109
110	110	111	111	112	113	114	114	114	114
115	115	116	116	117	118	119	119	119	119
120	120	121	121	122	123	124	124	124	124
125	125	126	126	127	128	129	129	129	129
130	130	131	131	132	133	134	134	134	134
135	135	136	136	137	138	139	139	139	139
140	140	141	141	142	143	144	144	144	144
145	145	146	146	147	148	149	149	149	149
150	150	151	151	152	153	154	154	154	154
155	155	156	156	157	158	159	159	159	159
160	160	161	161	162	163	164	164	164	164
165	165	166	166	167	168	169	169	169	169
170	NDFDIR	171	172	173	174	174	174	174	174
175	175	176	177	178	179	179	179	179	179
180	180	181	182	183	184	184	184	184	184
185	185	186	187	188	188	189	189	189	189
190	190	191	192	193	193	194	194	194	194
195	195	196	197	198	198	199	199	199	199
200	200	201	202	203	204	204	204	204	204
205	205	206	207	208	209	209	209	209	209
210	210	211	212	213	214	214	214	214	214
215	215	216	217	218	219	219	219	219	219
220	220	221	222	223	224	224	224	224	224
225	225	226	227	228	229	229	229	229	229
230	230	231	232	233	234	234	234	234	234
235	235	236	237	238	239	239	239	239	239
240	240	241	242	243	244	244	244	244	244
245	245	246	247	248	249	249	249	249	249
250	250	251	252	253	254	254	254	254	254
255	255								255

BEST AVAILABLE COPY

260	266	268	269	CRTK
265	271	273	274	
270	272	278	279	
275	276	CURTK		
280	281	282	283	
285	286	287	288	
290	291	292	293	PATLOC
295	296	297	298	
300	301	302	303	
305	306	307	308	
310	311	312	313	
315	316	317	318	
320	321	322	323	
325	326	327	328	
330	331	332	333	
335	336	337	338	
340	341	342	343	
345	346	347	348	
350	351	352	353	
355	356	357	358	
360	361	362	363	
365	366	367	368	
370	371	372	373	
375	376	377	378	
380	381	382	383	
385	386	387	388	
390	391	392	393	
395	396	PUTOUT	394	
400	401	UNENTS	399	
405	406	402	403	
410	411	407	408	TBLSZ
415	416	412	413	
420	421	417	418	FIXINT
425	426	422	423	
430	431	SCPNAME	424	
435	436	TT	428	TABNAM
440	441	432	433	
445	446	437	438	
450	451	442	443	BFOUT
455	456	447	448	
460	461	452	453	
465	466	457	458	
470	471	462	463	
475	476	467	468	
480	481	471	472	
485	NTOKNS	477	478	
490	ARDWNO	482	483	
495	LITLLOC	486	484	
500	501	487	488	
505	506	492	493	ALNLGH
510	511	497	498	
515	516	472	473	
520	521	471	472	
525	526	476	477	
		481	482	
		486	487	
		491	492	
		496	497	
		501	BFSET	
		506	507	
		511	512	
		516	517	
		518	519	
		522	523	
		526	527	
				SBXTNT

530	533	534
535	538	539
540	543	544
545	547	548
550	552	553
555	557	558
560	562	563
565	566	568
570	CRTKPS	571
575	CMSPN0	576
580	581	582
585	586	587
590	591	592
595	596	597
600	601	602
605	606	607
610	611	612
615	616	617
620	621	622
625	626	627
630	631	632
635	636	637
640	641	642
645	646	647
650	651	652
655	656	657
660	661	662
665	666	667
670	TSTRUCT	671
675	676	677
680	DCST2	681
685	686	687
690	691	692
695	696	697
700	701	702
705	706	707
710	711	712
715	716	717
720	721	722
725	726	727
730	731	732
735	736	737
740	741	742
745	746	747
750	EVANT	751
755	KNOLNM	756
760	761	762
765	766	767
770	LBIND	771
775	CBETB	776
780	781	782
785	786	787
790	791	792
795	796	797
		798
		799
		800
		801
		802
		803
		804
		805
		806
		807
		808
		809
		810
		811
		812
		813
		814
		815
		816
		817
		818
		819
		820
		821
		822
		823
		824
		825
		826
		827
		828
		829
		830
		831
		832
		833
		834
		835
		836
		837
		838
		839
		840
		841
		842
		843
		844
		845
		846
		847
		848
		849
		850
		851
		852
		853
		854
		855
		856
		857
		858
		859
		860
		861
		862
		863
		864
		865
		866
		867
		868
		869
		870
		871
		872
		873
		874
		875
		876
		877
		878
		879
		880
		881
		882
		883
		884
		885
		886
		887
		888
		889
		890
		891
		892
		893
		894
		895
		896
		897
		898
		899
		900
		901
		902
		903
		904
		905
		906
		907
		908
		909
		910
		911
		912
		913
		914
		915
		916
		917
		918
		919
		920
		921
		922
		923
		924
		925
		926
		927
		928
		929
		930
		931
		932
		933
		934
		935
		936
		937
		938
		939
		940
		941
		942
		943
		944
		945
		946
		947
		948
		949
		950
		951
		952
		953
		954
		955
		956
		957
		958
		959
		960
		961
		962
		963
		964
		965
		966
		967
		968
		969
		970
		971
		972
		973
		974
		975
		976
		977
		978
		979
		980
		981
		982
		983
		984
		985
		986
		987
		988
		989
		990
		991
		992
		993
		994
		995
		996
		997
		998
		999
		1000

800	SY4CLS	803	804
805	806	807	809
810	SYMTAB	812	814
815	816	817	819 TOK
820	821	822	823
825	826	827	828
830	831	832	833
835	836	837	838 DFSPT DCNEWD
840	841	842	843 DFSPTO
845	846	847	848 844
850	851	852	849 853 PRMTP
855	856	857	858 854
860	861	862	863 859
865	866	867	868 864
870	871	872	873 874
875	876	877	878 879
880	881	882	883 884
885	886	887	888 889
890	891	892	893 894
895	896	897	898 899
900	901	902	903 904
905	906	907	908 909
910	DEFNAM	912	913 914
911	SGNFLG	916	917 918
915	PRNT	920	921 922
920	925	926	927 928
925	930	931	932 PASSI
930	935	936	937 938
935	940	941	942 943
940	945	946	947 948
945	950	951	952 953
950	955	956	957 CURINT
955	960	961	962 963
960	965	966	967 968
965	970	971	972 973
970	975	976	977 978
975	980	981	982 SPOT HSHTAB
980	985	986	987 988
985	990	991	992 CMRDP
990	995	996	997 998
995	000	1001	1002 NDFCHR
000	005	1006	1007 1008
010	011	1011	1012 1013
015	016	1016	1017 1018
020	021	1021	1022 1023
025	026	1026	1027 1028
030	031	1031	1032 1033
035	036	1036	1037 1038
040	041	1041	1042 1043
045	046	1046	1047 1048
050	051	1051	1052 1053
055	056	1056	1057 1058
060	061	1061	1062 1063
065	066	1066	1067 SPTMP

BEST AVAILABLE COPY

1070		1074	
1075		1073	
1076		1077	
1080		1082	
1085		1087	
1090		1091	INDIC
1095	SZFLG	1096	BFTMP
1100		1097	NDFCAL
1105		1101	
1110		1106	STRG
1115		1111	
1120		1116	LITNAM
1125		1121	
1130		1126	TIP
1135		1131	TNWDSN
1140		1136	
1145		1141	HSCTST
1150		1146	
1155		1151	1147
1160		1156	IOSTAT
1165		1161	
1170		1166	TABOUT
1175		1167	ENSTRI
1180		1171	
1185		1176	TOKIN
1190		1177	
1195		1181	1178
1200		1182	
1205		1186	1183
1210		1191	
1215		1192	1184
1220		1195	1189
1225	DELIM	1196	1194
1230		1201	STCNT
1235	CBEPR	1206	
1240		1211	1193
1245		1216	1194
1250		1221	1195
1255		1225	1196
1260		1231	1197
1265		1236	1198
1270	ALDEFS	1241	1199
1275		1246	1204
1280		1247	1204
1285		1251	1209
1290		1256	
1295		1261	1173
1300		1266	1173
1305		1271	1173
1310		1276	1173
1315		1277	1173
1320		1278	1173
1325		1281	1173
1330	DEFNST	1286	1173
1335		1291	1173
1336		1296	1173
		1301	1173
		1302	1173
		1307	1173
		1311	1173
		1316	1173
		1317	1173
		1321	1173
		1322	1173
		1327	1173
		1331	1173
		1332	1173
		1333	1173
		1338	1173

1340	1341	1342	1343	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	LIKTAB	1386	1387	1388	1389	1389	1390	TISTCS	1391	1392	1393	1394	1395	CMSQN()	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	TYPTMP	1406	1407	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455	1456	1457	1458	1459	1460	ARDIMS	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471	1472	1473	1474	1475	1476	1477	1478	1479	1470	1471	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	DECTYP	1516	BFIRST	1517	1518	LIKCHR	1519	1520	1521	1522	1523	CUMOUT	1524	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534	1535	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551	PACFLG	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564	1565	1566	1567	1568	1569	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581	1582	1583	1584	1585	1586	1587	1588	1589	1590	1591	ARN0	1592	1593	1594	1595	1596	1597	1598	1599	1600	1601	CMTBDP	1602	1603	1604	1605
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------	------	------	------	------	------	------	--------	------	------	------	------	------	---------	------	------	------	------	------	------	------	------	------	------	--------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------	------	--------	------	------	--------	------	------	------	------	------	--------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------	------	------	------	------

1610	1612	1613	1614
1615	1616	1613	1619
1620	1621	1622	1624
1625	1626	1627	1629
1630	1631	1632	1634
1635	1636	1637	1639
1640	1641	1642	1644
1645	1646	1647	1649
1650	1651	1652	1654
1655	1656	1657	1659
1660	1661	1662	1664
1665	1666	1667	1669
1670	1671	1672	1674
1675	1676	1677	1679
1680	1681	1682	TOKPOS
1685	1686	1687	
1690	KNAME	1691	ONSTAT
1695	1696	1697	
1700	1701	1702	
1705	1706	1707	
1710	1711	1712	
1715	1716	1717	
1720	1721	1722	
1725	KNOPAT	1726	
1730	LBFLG	1731	
1735	LBFLG	1736	
1740	1741	1742	
1745	1746	1747	
1750	1751	1752	
1755	1756	1757	
1760	1761	1762	
1765	1766	1767	
1770	1771	1772	
1775	1776	1777	
1780	1781	1782	
1785	1786	1787	
1790	BFIN	1791	
1795		1796	
1800		1801	
1805	PRCSIN	1806	
1810		1811	
1815		1816	
1820		1821	
1825	TOKHSH	1826	
1830		1831	
1835		1836	
1840		1841	
1845		1846	
1850		1851	
1855		1856	
1860		1861	
1865		1866	
1870		1871	
1875		1876	
33			

1880	1881	1882	1883	1884
1885	1886	1887	1888	1889
1890	1891	1892	1893	1894
1895	1896	1897	1898	1899
1900	1901	1902	1903	1904
1905	1906	1907	1908	1909
1910	1911	1912	1913	1914
1915	1916	1917	1918	1919
1920	1921	1922	1923	1924
1925	1926	1927	1928	1929
1930	1931	1932	1933	1934
1935	1936	1937	1938	1939
1940	1941	1942	1943	1944
1945	TABBF	1946	1947	1948
1950	PITNAM	1951	1952	1953
1955	CBESB	1956	1957	1958
1960		1961	1962	1963
1965		1966	1967	1968
1970		1971	1972	1973
1975		1976	1977	1978
1980		1981	1982	1983
1985		1986	1987	1988
1990		1991	1992	1993
1995		1996	1997	1998

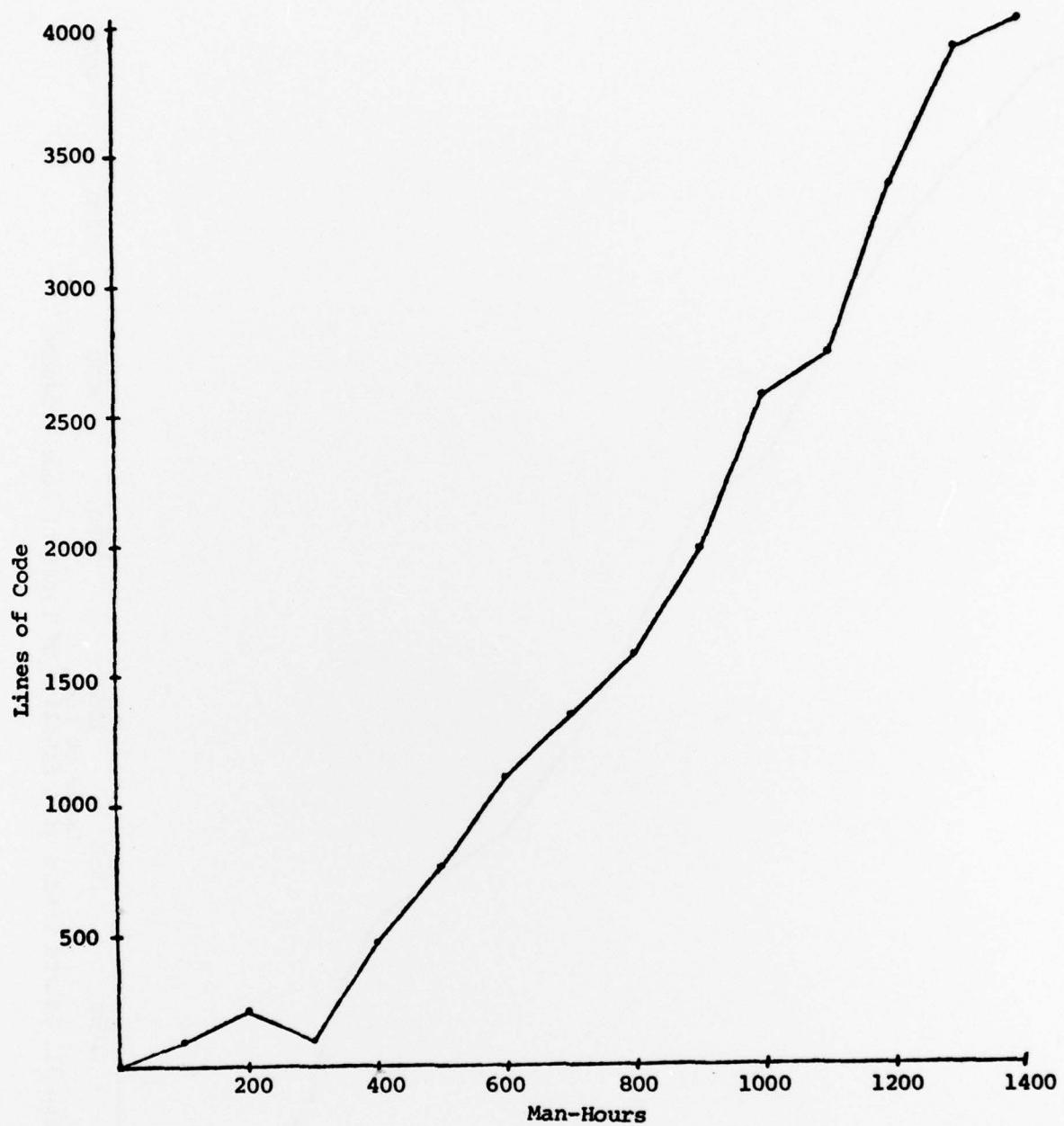
STCTMP ITDETR SYZ

SYMBOL TABLE

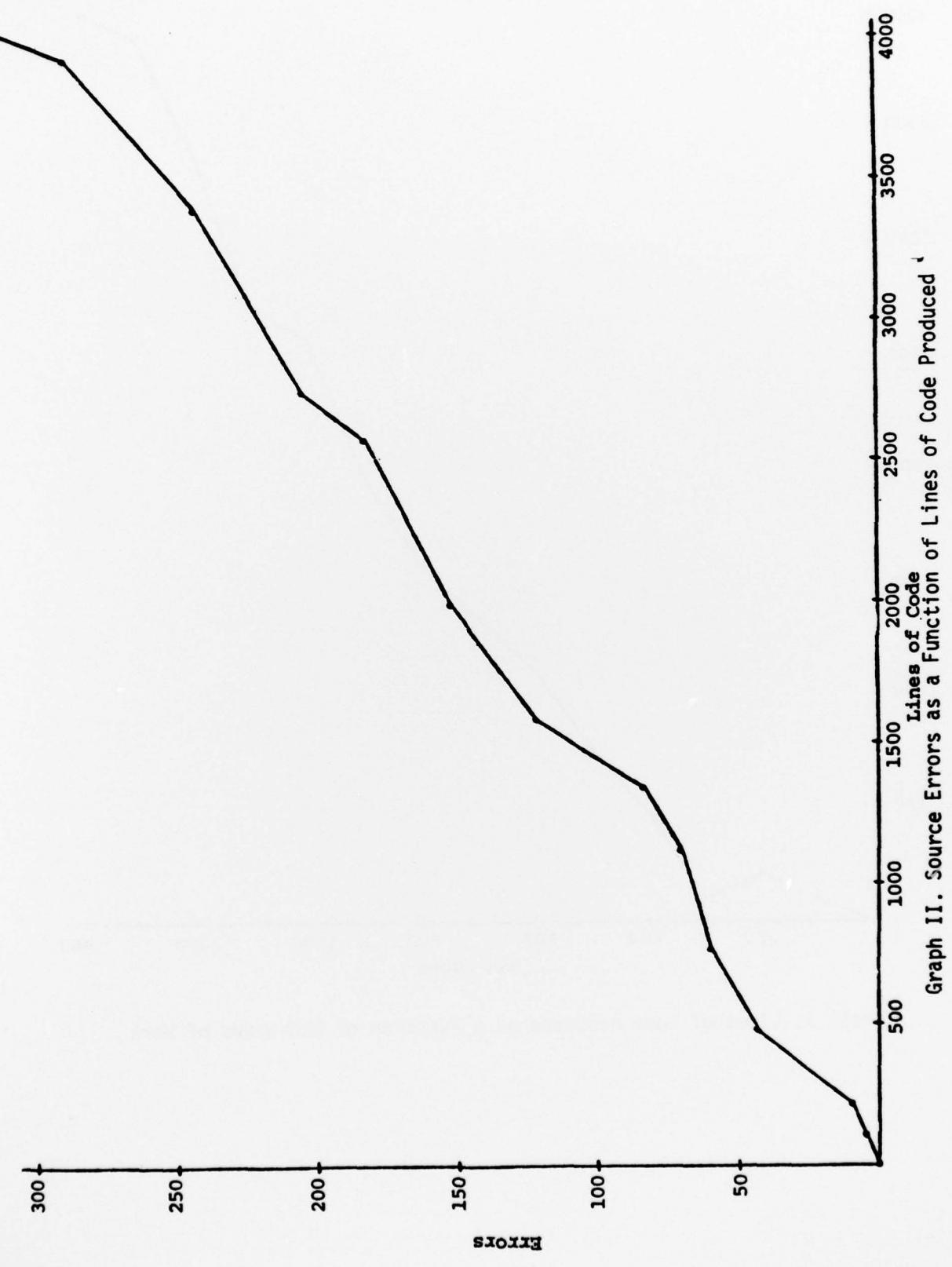
35

Table 11: Sample Statistics Output

LIKCHR	ITEM
GNUE	ITEM
PATLOC	ITEM
LITLOC	ITEM
HSCTST	ITEM
KNOPAT	ITEM
KNOLNM	ITEM
PROCEDURE	LIKTAB
H	B
	IS
	IS
	IS
	LABEL
	LABEL



Graph I. Lines of Code Produced as a Function of Man-hours of Work



Graph II. Source Errors as a Function of Lines of Code Produced